Amendments to the Claims

The listing of claims below will replace all prior versions and listings of claims in the present application.

Claim Listing

1   1-19. (Cancelled)

1   20. (Currently Amended) A circuit for checking the integrity of one or more
2   input vectors to a digital hardware block, comprising:
3   a memory array containing a weight matrix having elements which are based on a
4   set of known bad input vectors for the digital hardware block; and
5   means for selectively classifying future input vectors to the digital hardware block
6   as either good or not good, using the weight matrix, wherein the
7   classifying means classifies a given input vector as good or not good in
8   less than 60 ns;
9   an SRAM array;
10   a content-addressable memory for storing the known bad input vectors; and
11   address management means for using said SRAM to update both the content-
12   addressable memory and the weight matrix.

1   21. (Original) The circuit of Claim 20 further comprising means for creating the
2   weight matrix using a feedforward linear associative memory neural network.

1   22. (Original) The circuit of Claim 20 wherein said classifying means classifies
2   as not good both future input vectors which are definitely faulty and future inputs vectors
3   which are potentially faulty.

1   23. (Original) The circuit of Claim 20 wherein the weight matrix W is created
2   using a discrete Hopfield network algorithm according to the equation
3
4       $$w_{ij} = \sum_{m=1}^{M} (2a_i^{(m)} - 1)(2b_j^{(m)} - 1)$$
5

6   where $a^{(m)}$ is the set of known bad vectors, $a_i = b_j$, M is the number of bad input

7   vectors in the set of known bad input vectors, i is a row locator representing a particular

8   bad vector, and j is a column locator representing a bit location.

1   24. (Original) The circuit of Claim 23 wherein said classifying means includes

2   means for calculating an output vector $a^{(m)}$ by multiplying the weight matrix W by the

3   new input vector $b^{(m)}$, that is, $a^{(m)} = Wb^{(m)}$.

1   25. (Original) The circuit of Claim 24 wherein said classifying means further

2   includes means for adjusting elements of the output vector $a^{(m)}$ by its respective

3   thresholds θ_i according to the equation

4

5     $$\theta i = -\tfrac{1}{2} \sum_{j=1}^{K} w_{ij}$$

6

7

8   where K is the total number of bits in a vector.

1   26. (Original) The circuit of Claim 25 wherein said classifying means includes a

2   plurality of non-linear units which respectively processes the adjusted elements such that,

3   when a given adjusted element is positive, an output of the corresponding non-linear unit

4   is 1 and, when a given adjusted element is not positive, the output of the corresponding

5   non-linear unit is 0.

1   27. (Cancelled)

1   28. (Currently Amended) The circuit of Claim 20 further comprising ~~mean~~ means

2   for blocking an output of the digital hardware block corresponding to a new input vector,

3   and accepting an output of a software work-around corresponding to the new input

4   vector, in response to said classifying means classifying the new input vector as not good.

1   29. (Original) The circuit of Claim 20 further comprising means for updating the

2   weight matrix online using one or more additional bad input vectors.

1   30-32. (Cancelled)

1   33. (Original) A programmable logic unit comprising:
2   a plurality of interconnected hardware blocks, at least one of said hardware blocks
3   being a faulty hardware block;
4   a feedforward linear associative memory (LAM) neural network checking circuit
5   which classifies input vectors to said faulty hardware block as either good
6   or not good;
7   a software work-around input; and
8   a selection circuit connected to said feedforward LAM neural network checking
9   circuit, for (i) blocking an output vector of the faulty hardware block
10   corresponding to a new input vector, (ii) enabling a software work-around
11   for the new input vector, and (iii) accepting an output vector from said
12   software work-around input corresponding to the new input vector, as an
13   output vector of the programmable logic circuit.

1   34. (Original) The programmable logic unit of Claim 33 wherein said selection
2   circuit includes:
3   a software work-around enable signal output of said feedforward LAM neural
4   network checking circuit; and
5   a multiplexer having two inputs respectively receiving the output vector from said
6   faulty hardware block and the output vector from said software work-
7   around, and a select line which is connected to said software work-around
8   enable signal output of said feedforward LAM neural network checking
9   circuit.

1   35. (Original) The programmable logic unit of Claim 33 wherein said
2   feedforward LAM neural network checking circuit has a weight matrix whose elements
3   are based on a set of known bad input vectors for said faulty hardware block.

1   36. (Original) The programmable logic unit of Claim 35 wherein said

2   feedforward LAM neural network checking circuit updates the weight matrix online

3   using one or more additional bad input vectors.

1   37. (Original) The programmable logic unit of Claim 36 wherein said

2   feedforward LAM neural network checking circuit includes:

3   an SRAM array;

4   a content-addressable memory for storing the known bad input vectors; and

5   address management means for using said SRAM to update both said content-

6   addressable memory and said weight matrix.

1   38. (Original) The programmable logic unit of Claim 35 wherein the weight

2   matrix W in said feedforward LAM neural network checking circuit is calculated

3   according to the equation

4
5      $w_{ij} = \sum_{m=1}^{M} (2a_i^{(m)} - 1)(2b_j^{(m)} - 1)$
6

7   where $a^{(m)}$ is the set of known bad vectors, $a_i = b_j$, M is the number of bad input

8   vectors in the set of known bad input vectors, i is a row locator representing a particular

9   bad vector, and j is a column locator representing a bit location.

1   39. (Original) The programmable logic unit of Claim 38 wherein said

2   feedforward LAM neural network checking circuit includes:

3   means for calculating an output vector $a^{(m)}$ by multiplying the weight matrix W by

4   the new input vector $b^{(m)}$, that is, $a^{(m)} = Wb^{(m)}$;

5   means for adjusting elements of the output vector $a^{(m)}$ by its respective thresholds

6   θ_i according to the equation

7
8      $\theta i = -\frac{1}{2} \sum_{j=1}^{K} w_{ij}$
9
10

11   where K is the total number of bits in a vector; and

12   a plurality of non-linear units which process respective elements of the weight

13   matrix to provide an output of 1 when a given adjusted element is positive,

14   and provide an output of 0 when a given adjusted element is not positive.

1   40. (Original) The programmable logic unit of Claim 39 wherein said

2   feedforward LAM neural network checking circuit matches a vector constructed of the

3   outputs of said non-linear units with an entry in a content-addressable memory storing the

4   set of known bad vectors.

1   41. (Original) The programmable logic unit of Claim 33 wherein said

2   feedforward LAM neural network checking circuit classifies input vectors to said faulty

3   hardware block as either good or not good prior to said faulty hardware generating its

4   output vector corresponding to the new input vector.

1   42. (Original) The programmable logic unit of Claim 33 wherein said

2   feedforward LAM neural network checking circuit is physically located in a re-

3   configurable hardware redundancy block.

        Serial No.: 09/837,882